

Spotfire® Cobranding

Software Release 14.4.1

Contents

Cobranding Spotfire.....	4
Creating a cobranding.config file.....	5
The cobranding.config template.....	5
Replacing the product name in Spotfire Analyst.....	8
Replacing the banner in Spotfire Analyst.....	9
Replacing the image in the login dialog for Spotfire Analyst.....	10
Customizing the landing page in Spotfire Analyst and Web Player clients.....	11
The landingPageTemplate.html template.....	11
Customizing the toolbar shortcuts for Spotfire Analyst and Web Player clients.....	18
The defaultPreferences.json template.....	18
Replacing the Support Information in Spotfire Analyst and Spotfire Web Player.....	20
Replacing the Spotfire Community option in Spotfire Analyst and Spotfire Web Player.....	21
Replacing the "About" information in Spotfire Analyst and Spotfire Web Player.....	22
Replacing the product name in the Spotfire Server web UI.....	23
Adding custom headers to Spotfire UIs.....	24
The WebHeader template.....	25
Replacing the default Login page in the Spotfire Server web UI.....	27
The custom-login-app-example.html template.....	28
The custom-login-app-example.js template.....	29
The Custom Login Page API.....	30
The custom-login-app-example.css template.....	31
Creating and deploying a cobranding package.....	33
Spotfire Documentation and Support Services.....	35

Legal and Third-Party Notices..... 36

Cobranding Spotfire

Cobranding Spotfire, or white labeling it, means that you customize its look and feel to your own company profile. You can rebrand Spotfire®, your Spotfire® Web Player instances, and the Spotfire® Server web administration pages by replacing the product name, logo, and similar resources. Information in the Support, About and Community menu entries under Help can be replaced, and you can specify which toolbar shortcuts your end users should see by default. You can also customize the landing page, and add a custom login page for the Spotfire Server web interface.

The custom logotype images and names are specified directly in the `cobranding.config` XML file. Other files are also needed for some customizations, see the corresponding section.



For cobranding to work on a Linux system, cabextract must be installed.

The configuration file, together with other custom files or images are then deployed as a package to the server. The changes to Spotfire® Analyst and the server and web player interfaces are visible the next time that users connect to the server.

Creating a cobranding.config file

To replace the product name, banner, header, logo, and other items in the Spotfire interface, you must have a cobranding.config file.

Procedure

1. Copy the [cobranding template](#) to an XML editor or text editor and save the template as an XML file with the file name cobranding.config.
2. Place the file in a local directory that will contain all of your cobranding files.

The cobranding.config template

You can copy and edit this cobranding.config template to create an XML file for your cobranded version of Spotfire.

```
<?xml version="1.0" encoding="utf-8"?>
<cobranding>
  <!--
    When specifying a URL, the following characters must be escaped:

    Character      Escape code
    -----
    SPACE          %20
    <              %3C
    >              %3E
    #              %23
    %              %25
    {              %7B
    }              %7D
    |              %7C
    \              %5C
    ^              %5E
    ~              %7E
    [              %5B
    ]              %5D
    `              %60
    ;              %3B
    /              %2F
    ?              %3F
    :              %3A
    @              %40
    =              %3D
    &              %26
    $              %24

    The only valid schemes to use in the URL are HTTP and HTTPS. If a scheme
    is not provided, the scheme is assumed to be HTTP.

    To assign an access key to a menu item, the letter intended
    to be used as access key must be prefixed with an ampersand (&). This
    character also must be XML encoded to &amp;. To display an actual
    ampersand in these items, two escaped ampersand characters must be
    placed next to each other (&amp;&amp;).

    All images used in the cobranding package must be in the PNG format. If
    the images are in a different format, the cobranding package uses the
    default images in Spotfire.
  -->

  <!--
    Replaces product name in window title and login dialog of Spotfire Analyst.
    If the product name exceeds the available space in the login dialog, the
    product name is trimmed.
  -->
  <productName>Sample Product Name</productName>
```

```

<!--
  Replaces product name in window title, login dialog, and About page for the
  Web Player. If the product name exceeds the available space in the login
  dialog, the product name is trimmed.
-->
<productNameWeb>Sample Product Name</productNameWeb>

<!--
  The banner is used in splash screens and the About dialog of Spotfire Analyst.
  The banner should be 700 pixels x 57 pixels, and able to be safely cropped to
  400 pixels x 57 pixels. The cropping is done on the right side of the image.
  Vital image information should be placed on the left side of the image to
  ensure that it will not disappear when cropped.
-->
<logoBanner>banner.png</logoBanner>

<!--
  Replaces the "Spotfire Support" text on the Help menu and the support
  link. If the support URL exceeds the available space in the About dialog,
  the link is trimmed and the full URL is displayed in a tooltip.
-->
<support>
  <text>Sample & Support Link</text>
  <url>www.example.com/support</url>
</support>

<!--
  Replaces the "Spotfire community" text on the Help menu, and the link to the
  Spotfire community.
-->
<community>
  <text>Spotfire Community</text>
  <url>https://spotfi.re/community/</url>
</community>

<!--
  Replaces the "About Spotfire" text on the Help menu.
-->
<about>Sample & About Box</about>

<!--
  Link in the bottom left of the About dialog.
-->
<companyUrl>www.example.com</companyUrl>

<web>
<!--
  The WebHeader.htm file is used to add a custom header to the server UI
  and Web Player.
-->
<header>WebHeader.htm</header>

<!--
  The logo shown in the navigation header on the library page. If the
  image exceeds the dimensions 190x33 pixels, the image will be cropped.
-->
<navigationLogo>libraryNavigationLogo.png</navigationLogo>
</web>

<!--
  To override the Spotfire default toolbar configuration without changing
  the group preference, supply a json-file that contains a json object with
  the name "ToolbarShortcuts" and the value is an array of toolbar identifiers.
-->
<defaultPreferences>defaultPreferences.json</defaultPreferences>

<!--
  The landing page shown when you open Spotfire, after you have logged in.
  Supply a .html file with the content and styling that you
  want to use as the landing page.
-->
<landingPage>LandingPageTemplate.html</landingPage>

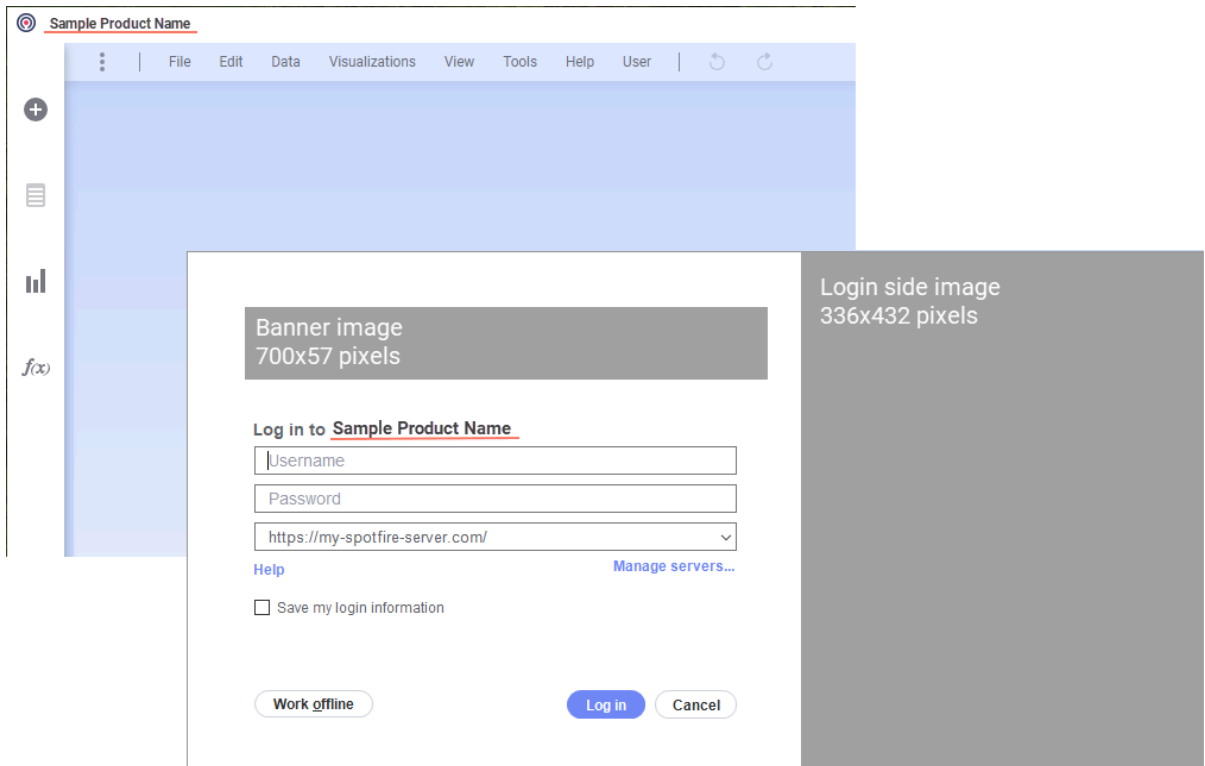
<!--
  The image shown on the right side of the Spotfire Analyst login dialog.
  The image dimensions should be 336 x 432 pixels. If the image dimensions

```

```
are above or below 336 x 432 pixels, the image is resized.  
-->  
<loginPageSideImage>loginSideImage.png</loginPageSideImage>  
</cobranding>
```

Replacing the product name in Spotfire Analyst

You can replace the product name in the login screen and window titles of Spotfire.



Prerequisites

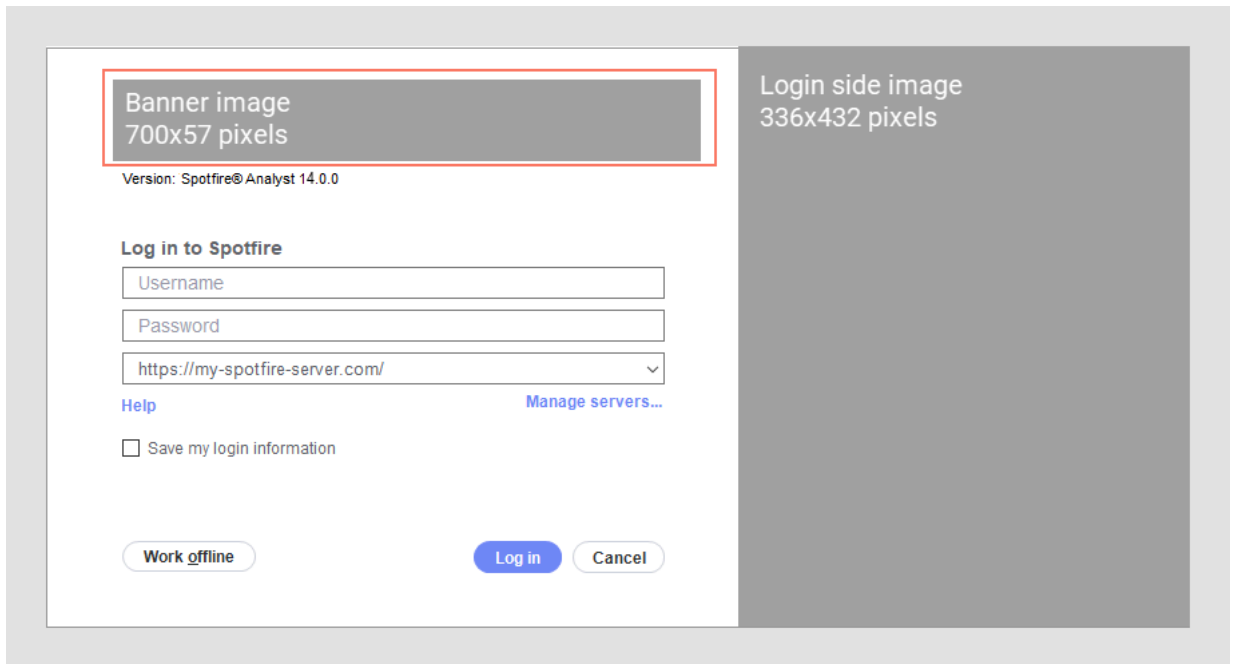
You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the section `<productName>Sample Product Name</productName>`.
3. Replace "Sample Product Name" with the product name you want to use in the cobranded version of Spotfire Analyst. If the product name exceeds the available space in the login dialog, the product name will be trimmed.

Replacing the banner in Spotfire Analyst

You can replace the banner in the startup splash screen, login screen, and the About dialog of Spotfire Analyst.



Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

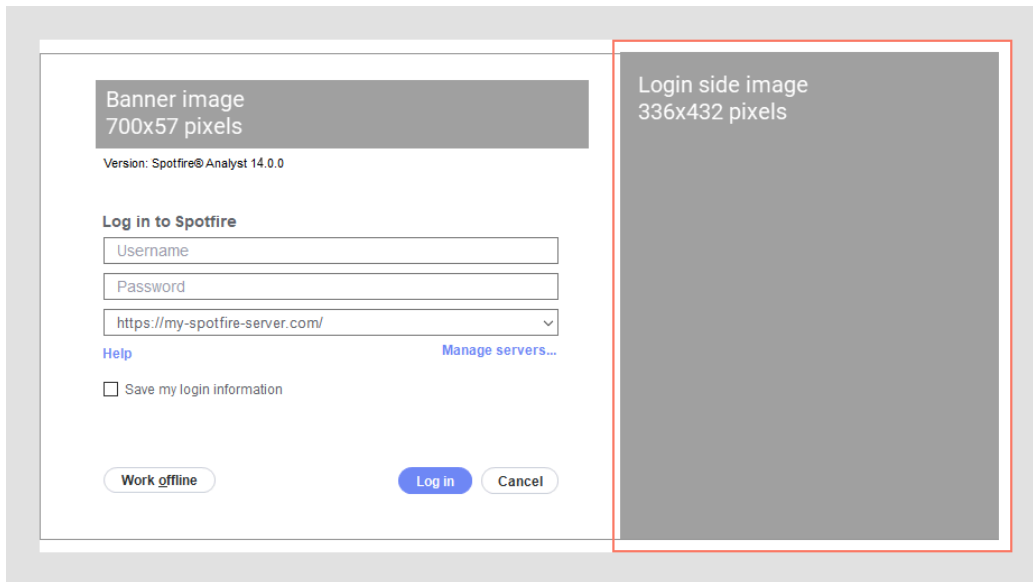
1. Create a banner to use in Spotfire. The banner must have these properties:
 - It is a .png file.
 - The dimensions do not exceed 700x57 pixels.
 - The maximum width is 700 pixels. If the width exceeds 700 pixels, the image is cropped from the right.
 - The maximum height is 57 pixels. If the height exceeds 57 pixels, the image is resized.
2. Open the `cobranding.config` file in an XML editor or text editor.
3. Locate the section `<logoBanner>banner.png</logoBanner>`.
4. Make sure that the name in the `<logoBanner>` section matches the name of the image that you want to use as a banner.
5. Add the image to the local folder where the `cobranding.config` file is located.

Replacing the image in the login dialog for Spotfire Analyst

You can change the image on the right side of the login dialog for Spotfire Analyst.

Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).



Procedure

1. Create an image file that you want to use as the side image in the login dialog for Spotfire Analyst. The image must have these properties:
 - It is a .png file.
 - The dimensions are exactly 336x432 pixels. If the image is larger or smaller, the image is resized.
2. Save the image to the local folder where the `cobranding.config` file is located.
3. Open the `cobranding.config` file in an XML editor or text editor.
4. Locate or add the setting `loginPageSideImage`, and make sure it points to your login image .png file.

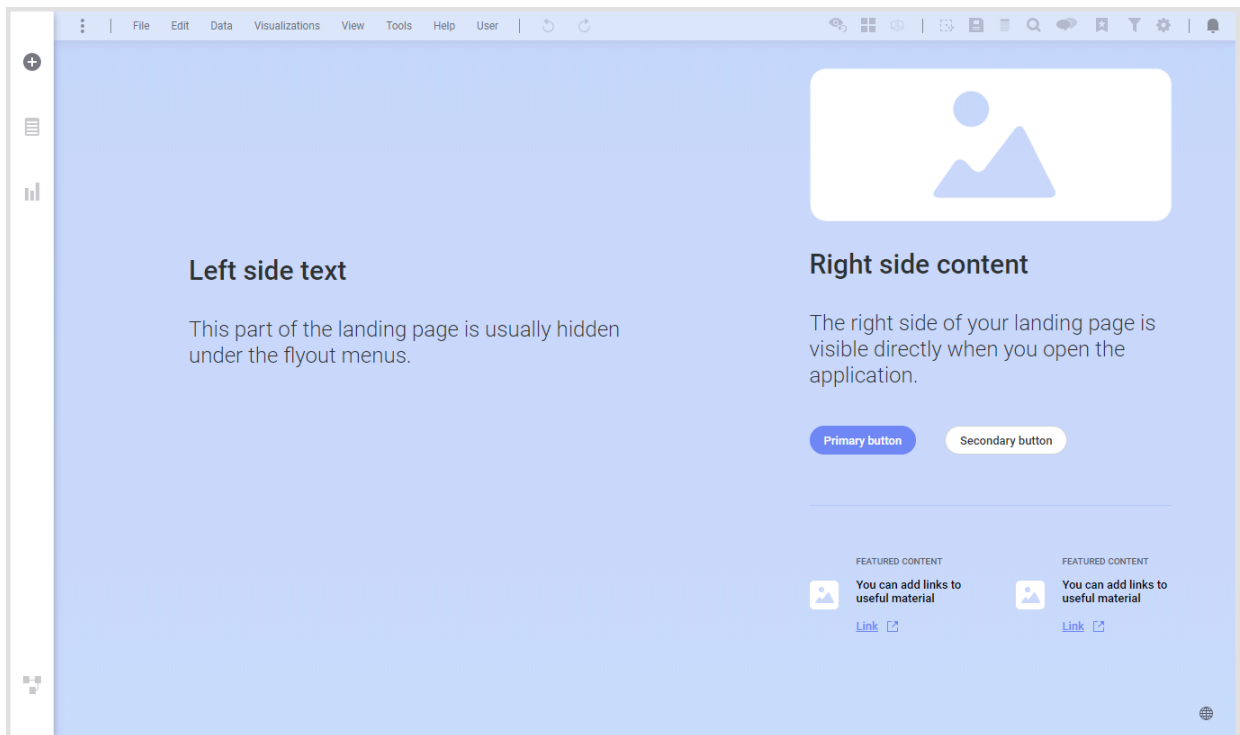
```
<loginPageSideImage>loginSideImage.png</loginPageSideImage>
```

Customizing the landing page in Spotfire Analyst and Web Player clients

You can replace and customize the contents on the landing page that you see when you open a Spotfire client, after you have logged in. To create your own landing page, you use an `.html` file. With HTML and CSS you have rich options for making the landing page your own.

Prerequisites

You need a `cobranding.config` file to work with. To create one, see [Creating a cobranding.config file](#).



Procedure

1. Create an `.html` file that you want to use for the landing page.



To quickly get started, you can copy and modify the included [HTML template](#).

2. Save the `.html` file to the directory where your `cobranding.config` file is located.



In the `cobranding.config` template, the default filename of the `.html` file for the landing page is `LandingPageTemplate.html`.

3. Open the `cobranding.config` file in an XML or text editor.
4. Locate the setting `landingPage`, and make sure the filename matches your landing page `.html` file.

```
<landingPage>LandingPageTemplate.html</landingPage>
```

5. Save your changes.

The landingPageTemplate.html template

To customize the landing page in the Spotfire clients, you use an `.html` file. In the `cobranding.config` template, the file is named `landingPageTemplate.html`.

Guidelines

The following are rules and tips to keep in mind when you write your own landing page.

Links

The attribute `target` in for links is automatically set to `_blank`.

You cannot use other values for the `target` attribute.

Images

- To use a locally stored image file, embed the file in the cobranding package. Then in the `.html` file refer to the image file starting the variable name `[%AppPath%]`. For example:

```
[%AppPath%]/myImage.png
```

- To use an image stored on the web, refer to the image with an absolute URL. For example:

```
http://example.com/myImage.png
```

Sample landing page template

This is a sample customized landing page HTML template, that you can copy and modify to get started with your own landing page.

```
<html>
<head>
  <link
    href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;500;700&display=swap"
    rel="stylesheet"
  />

  <style>
    .customLandingPageTemplate {
      height: 100%;
      display: flex;
      overflow: hidden;
    }

    .leftSide {
      display: flex;
      flex-direction: column;
      align-items: center;
      min-width: 768px;
    }

    .leftSideContent {
      display: flex;
      flex-direction: column;
      max-width: 500px;
      flex-basis: 47%;
      justify-content: flex-end;
    }

    .rightSide {
      display: flex;
      justify-content: center;
      flex: 1;
    }

    .rightSideContent {
      display: flex;
      flex-direction: column;
      max-width: 402px;
    }

    .textContainer {
      display: flex;
      flex-direction: column;
      flex-basis: 25%;
      justify-content: flex-end;
    }

    .primaryText {
      font: 30px "Roboto";
```

```

    font-weight: 500;
    margin-bottom: 32px;
    color: #313336;
  }

  .secondaryText {
    font: 24px "Roboto";
    font-weight: 300;
    color: #313336;
  }

  .imageContainer {
    flex-basis: 25%;
    display: flex;
  }

  .frame {
    display: flex;
    align-items: flex-end;
  }

  .buttonsContainer {
    align-items: flex-end;
    display: flex;
    flex-basis: 10%;
  }

  .button {
    height: 32px;
    border: none;
    padding: 0px 16px;
    cursor: pointer;
    border-radius: 16px;
    font: 13px "Roboto";
    font-weight: 500;
  }

  .button:focus {
    outline: 0;
  }

  .primaryButton {
    background-color: #7289f9;
    color: white;
    margin-right: 32px;
  }

  .primaryButton:hover {
    background-color: #5d6efa;
  }

  .primaryButton:active {
    background-color: #3050ef;
    border: none;
  }

  .secondaryButton {
    background-color: white;
    color: #313336;
    border: 1px solid #d1d4de;
  }

  .secondaryButton:hover {
    background-color: #edf0f7;
  }

  .secondaryButton:active {
    background-color: #ebefff;
    color: #3050ef;
  }

  .horizontalLineContainer {
    display: flex;
    align-items: center;
    height: 112px;
  }

  .horizontalLine {
    flex: 1;
    background: #b8c8fd;
  }

```

```

        height: 1px;
    }

    .menuItemsContainer {
        display: flex;
    }

    .leftMenuItem {
        flex: 0.5;
        display: flex;
        margin-right: 56px;
    }

    .rightMenuItem {
        flex: 0.5;
        display: flex;
    }

    .menuIcon {
        flex: 0.3;
        display: flex;
        align-items: center;
    }

    .menuText {
        flex: 0.7;
        display: flex;
        flex-direction: column;
        justify-content: center;
    }

    .featuredContent {
        font: 10px "Roboto";
        font-weight: 500;
        color: #61646b;
        margin-bottom: 12px;
    }

    .linkText {
        margin-bottom: 16px;
        font: 13px "Roboto";
        font-weight: 500;
    }

    .linkStyle {
        color: #7289f9;
        display: flex;
        font: 13px "Roboto";
        font-weight: 500;
    }

    .linkStyle:hover {
        fill: #5d6efa;
        color: #5d6efa;
    }

    .linkStyle:active {
        fill: #3050ef;
        color: #3050ef;
    }

    .linkIcon {
        fill: #7289f9;
        margin-left: 8px;
    }

    .linkStyle:hover svg {
        fill: #5d6efa;
    }

    @media screen and (max-width: 900px) {
        .leftSide {
            display: none;
        }
    }
</style>
</head>
<body>
    <div class="customLandingPageTemplate">
        <!-- Left Side -->

```

```
<div class="leftSide">
  <div class="leftSideContent">
    <div class="textContainer">
      <div class="primaryText">
        <label>Left side text</label>
      </div>

      <div class="secondaryText">
        <label>
          >This part of the landing page is usually hidden under the flyout
            menus.</label>
        >
      </div>
    </div>
  </div>
</div>
<!-- End Left Side -->

<!-- Right Side -->
<div class="rightSide">
  <div class="rightSideContent">
    <div class="imageContainer">
      <div class="frame">
        
      </div>
    </div>

    <div class="textContainer">
      <div class="primaryText">
        <label>Right side content</label>
      </div>

      <div class="secondaryText">
        <label>
          >The right side of your landing page is visible directly when
            you open the application.</label>
        >
      </div>
    </div>
  </div>

  <!-- Buttons & Links Examples -->
  <div class="buttonsContainer">
    <button class="button primaryButton"
      onclick="window.open('https://spotfi.re',
        '_blank');">Primary button</button>
    <button class="button secondaryButton"
      onclick="window.open('https://spotfi.re',
        '_blank');">Secondary button</button>
  </div>

  <div class="horizontalLineContainer">
    <div class="horizontalLine"></div>
  </div>

  <div class="menuItemsContainer">
    <div class="leftMenuItem">
      <div class="menuItemIcon">
        <!-- SVG example -->
        <svg
          xmlns="http://www.w3.org/2000/svg"
          width="32"
          height="32"
          viewBox="0 0 32 32"
        >
          <path
            d="M27.32,0H4.681A4.681,4.681,0,0,0,0,4.681,0,0,0,4.681,
              4.681V27.319A4.681,4.681,0,0,0,4.681,32H27.32A4.68,4.68,0,0,0,32,27.319V4.681A4.68,
              4.68,0,0,0,27.32,0ZM13.7a3,3,0,1,1-3,3A3,3,0,
              0,1,13,7ZM26.017,25H6.882c-.84,
              0-1.13-.544-.65-1.216l3.325-4.632a1.24,1.24,0,0,
              1,1.941-.161l1.284,1.268a1.307,1.307,0,0,0,
              1.993-.123l5.249-6.689a.887.887,0,0,1,
              1.609,1.461l5.219,10.082C27.231,24.407,26.857,
              25,26.017,25Z"
              fill="#fff"
            />
          </svg>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

<div class="menuText">
  <div class="featuredContent">FEATURED CONTENT</div>

  <div class="linkText">You can add links to useful material</div>
  <a href="https://spotfi.re/community"
  target="_blank" class="linkStyle">
    Link
    <svg
      class="linkIcon"
      xmlns="http://www.w3.org/2000/svg"
      width="16"
      height="16"
      viewBox="0 0 16 16"
    >
      <path
        d="M7,2V3H3V13H13V9h1v4.25a.749.749,0,
        0,1-.75.75H2.75A.75.75,0,0,1,2,
        13.25V2.75A.75.75,0,0,1,2.75,2Z"
      />

      <polygon points="14 7 9 2 14 2 14 7" />

      <rect
        x="6.252"
        y="5.505"
        width="7.495"
        height="0.99"
        transform="translate(-1.314 8.828) rotate(-45)"
      />
    </svg>
  </a>
</div>
</div>

<div class="rightMenuItem">
  <div class="menuIcon">
    <svg
      xmlns="http://www.w3.org/2000/svg"
      width="32"
      height="32"
      viewBox="0 0 32 32"
    >
      <path
        d="M27.32,0H4.681A4.681,4.681,0,0,0,0,
        4.681V27.319A4.681,4.681,0,0,0,4.681,
        32H27.32A4.68,4.68,0,0,0,32,27.319V4.681A4.68,
        4.68,0,0,0,27.32,0ZM13,7a3,3,0,1,1-3,3A3,3,0,
        0,1,13,7ZM26.017,25H6.882c-.84,
        0-1.13-.544-.65-1.216l3.325-4.632a1.24,1.24,0,0,
        1,1.941-.161l1.284,1.268a1.307,1.307,0,0,0,
        1.993-.123l5.249-6.689a.887.887,0,0,1,
        1.609.146l5.219,10.082C27.231,24.407,26.857,
        25,26.017,25Z"
        fill="#fff"
      />
    </svg>
  </div>

  <div class="menuText">
    <div class="featuredContent">FEATURED CONTENT</div>

    <div class="linkText">You can add links to useful material</div>

    <a href="https://spotfi.re/community"
    target="_blank" class="linkStyle"
    >Link

    <svg
      class="linkIcon"
      xmlns="http://www.w3.org/2000/svg"
      width="16"
      height="16"
      viewBox="0 0 16 16"
    >
      <path
        d="M7,2V3H3V13H13V9h1v4.25a.749.749,0,
        0,1-.75.75H2.75A.75.75,0,0,1,2,
        13.25V2.75A.75.75,0,0,1,2.75,2Z"
      />

```



```
<polygon points="14 7 9 2 14 2 14 7" />

<rect
  x="6.252"
  y="5.505"
  width="7.495"
  height="0.99"
  transform="translate(-1.314 8.828) rotate(-45)"
/>
</svg>
</a>
</div>
</div>
</div>
</div>
<!-- End Right Side -->
</div>
</body>
</html>
```

Customizing the toolbar shortcuts for Spotfire Analyst and Web Player clients

You can define a default set of shortcuts for the toolbar in Spotfire Analyst and Web Player clients.



Prerequisites

You need a `cobranding.config` file to work with. To create one, see [Creating a cobranding.config file](#).

Procedure

1. If it is not already available in your cobranding package, create a [defaultPreferences.json](#) file to add preferences.
2. Edit the `defaultPreferences.json` file in a text editor, and add the toolbar shortcuts you want in the "ToolbarShortcuts" setting:

```
"ToolbarShortcuts": [
  "Spotfire.ResetVisibleFilters",
  "Spotfire.ArrangeVisualizationsEvenly",
  "Spotfire.ScatterPlot3D",
  "Spotfire.VerticalLine",
  "Spotfire.DetailsOnDemandPanel",
  "Spotfire.Save",
  "Spotfire.Table",
  "Spotfire.Find",
  "Spotfire.Collaboration",
  "Spotfire.BookmarkPanel",
  "Spotfire.FilterPanel",
  "Spotfire.VisualizationProperties",
  "Spotfire.VerticalLine"
]
```



For the full list of available toolbar shortcuts and their names, see the list under **Preference descriptions > Application preferences > ToolbarShortcuts** in the documentation for [Spotfire Administration Manager](#).

3. Open the `cobranding.config` file in an XML or text editor.
4. Locate or add the setting `defaultPreferences`, and make sure it points to your preferences .json file.

```
<defaultPreferences>defaultPreferences.json</defaultPreferences>
```

Result

The set of shortcuts that you define in your cobranding package becomes the default toolbar layout for the Spotfire Analyst installed client, and the web clients. Users who want to add their own set of shortcuts can easily [customize the toolbar in the Spotfire client](#).

The defaultPreferences.json template

To define preferences such as a default set of shortcuts for the custom toolbar in Spotfire clients, you use a .json file. By default, the name of the file used for this is `defaultPreferences.json`. This is an example that you can copy and paste into your own .json preferences file.

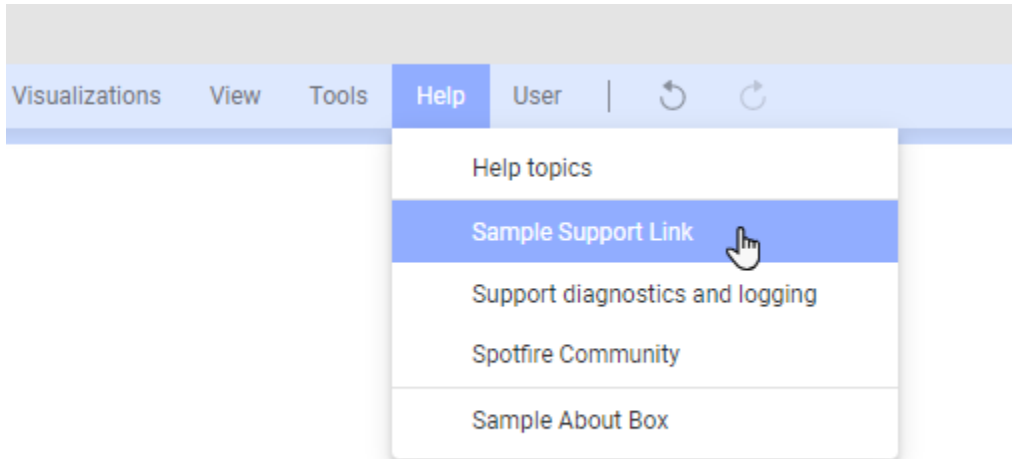


The specific toolbar shortcuts in the list below are only included as an example. For the full list of toolbar shortcuts and their names, see the list under **Preference descriptions > Application preferences > ToolbarShortcuts** in the documentation for [Spotfire Administration Manager](#).

```
{
  "ToolbarShortcuts": [
    "Spotfire.ResetVisibleFilters",
    "Spotfire.ArrangeVisualizationsEvenly",
    "Spotfire.ScatterPlot3D",
    "Spotfire.VerticalLine",
    "Spotfire.DetailsOnDemandPanel",
    "Spotfire.Save",
    "Spotfire.Table",
    "Spotfire.Find",
    "Spotfire.Collaboration",
    "Spotfire.BookmarkPanel",
    "Spotfire.FilterPanel",
    "Spotfire.VisualizationProperties",
    "Spotfire.VerticalLine"
  ]
}
```

Replacing the Support Information in Spotfire Analyst and Spotfire Web Player

You can replace the "Spotfire Support" text on the **Help** menu and the URL behind it, as well as the link that follows the text "For support, please contact" in the About dialog. These changes affect both Spotfire Analyst and the Spotfire Web Player.



Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

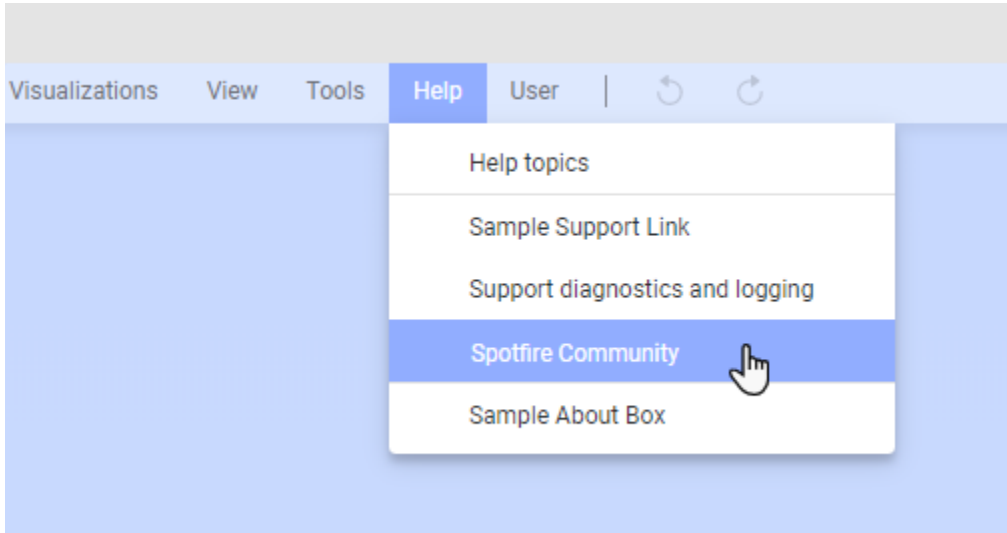
1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the following section:

```
<support>
  <text>Sample &Support Link</text>
  <url>www.example.com/support</url>
</support>
```

3. Replace "Sample Support Link" with the text that you want to use on the **Help** drop-down menu.
4. To select a letter as the access key for this menu item, type `&` in front of the letter.
5. Replace "www.example.com/support" with the support URL that you want to use for the link. This URL will also appear in the About dialog.

Replacing the Spotfire Community option in Spotfire Analyst and Spotfire Web Player

You can replace the text for the "Spotfire Community" option on the **Help** menu, and the URL behind it. These changes affect both Spotfire Analyst and the Spotfire Web Player.



If you want, you can decide not to display the **Spotfire Community** option at all in the Help drop-down menu, rather than replacing it. Use the preference setting `ShowCommunity` in the Spotfire Administration Manager. Set the value of the preference to `False` for the user groups that you want not to see the **Spotfire Community** option.

Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

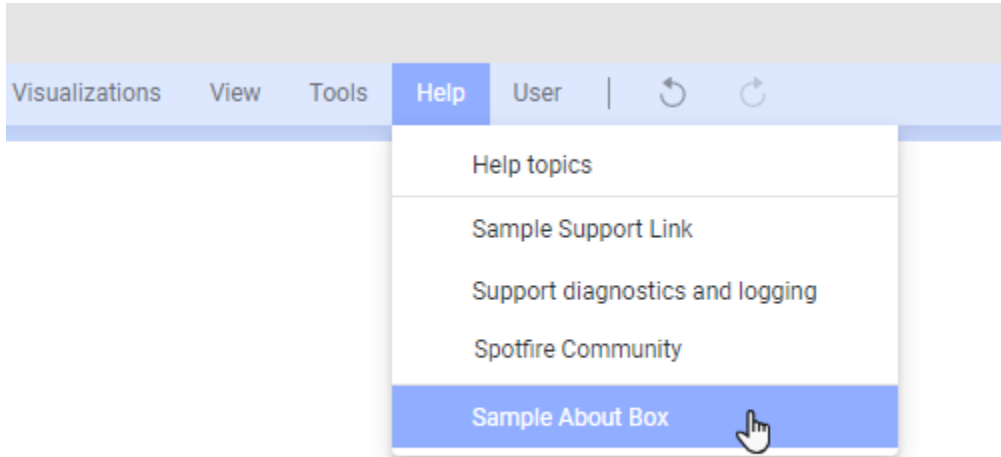
1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the following section:

```
<community>
  <text>Spotfire Community</text>
  <url>https://spotfi.re/community</url>
</community>
```

3. Replace "Spotfire Community" with the text that you want to use on the **Help** drop-down menu.
4. To select a letter as the access key for this menu item, type `&#x26;` in front of the letter.
5. Replace "https://spotfi.re/community" with the community URL that you want to use for the link.

Replacing the "About" information in Spotfire Analyst and Spotfire Web Player

You can replace the "About Spotfire" text on the **Help** menu, the title of the About dialog, and the URL in the About dialog. These changes affect both Spotfire Analyst and Spotfire Web Player.



Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the following lines:

```
<about>Sample &Amp;About Box</about>
```

```
<companyUrl>www.example.com</companyUrl>
```

3. Replace "Sample About Box" with the text that you want to use on the menu and as the title of the About dialog. To select a letter as the access key for the menu item, type `&Amp;` in front of the letter.

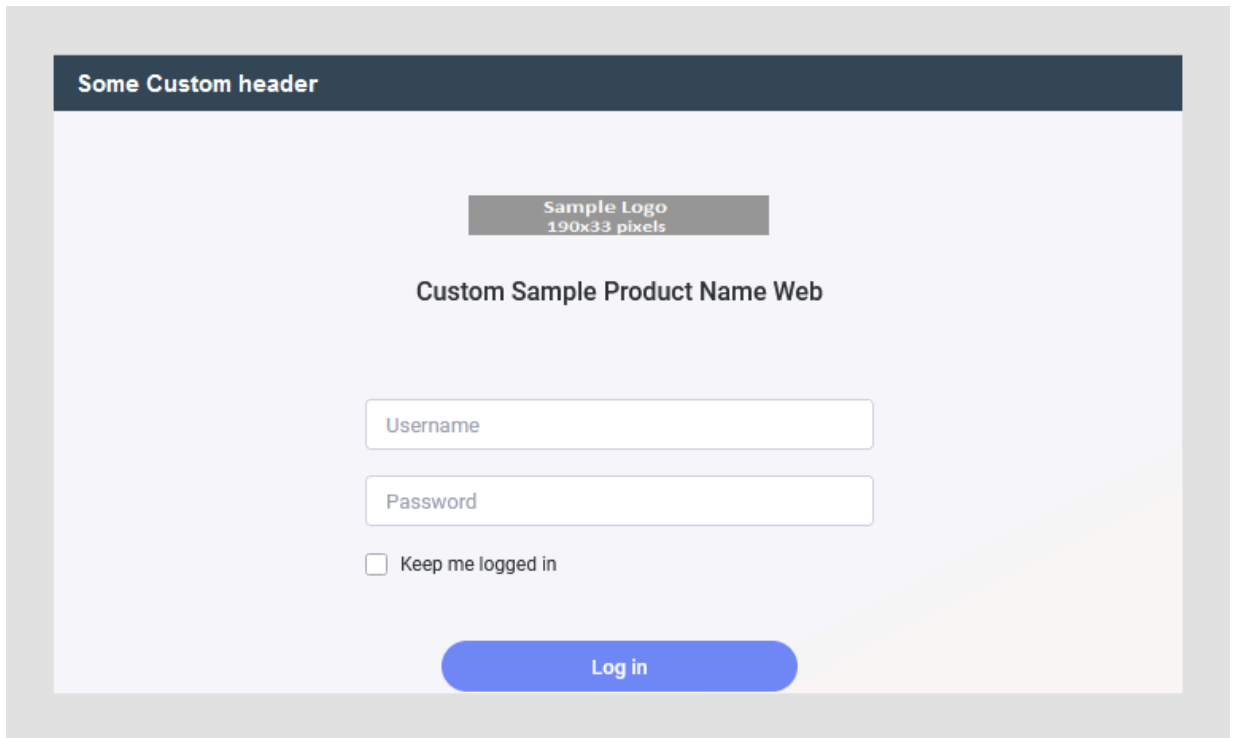


In the web client, the title of the dialog is specified by [the product name](#).

4. Replace "www.example.com" with the URL that you want to use in the lower-left corner of the About dialog.

Replacing the product name in the Spotfire Server web UI

You can replace the product name on the login screen, and other places where the product name is shown in the web clients and the Server administration user interface.



Prerequisites

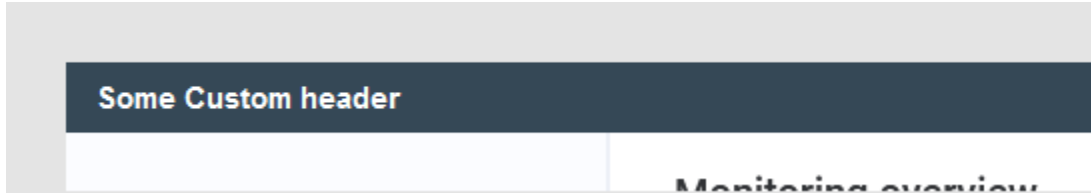
You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the section `<productNameWeb>Sample Product Name</productNameWeb>`.
3. Replace "Sample Product Name Web" with the product name that you want to use in the co-branded version of the web client. If the product name exceeds the available space, the product name will be trimmed.

Adding custom headers to Spotfire UIs

You can use your own header in the Spotfire UIs. If you include the custom header in the cobranding package, and add the cobranding package to the default deployment area, the header will be displayed in the Spotfire Server UIs and the web player UIs. No restart or update is required, but you might need to refresh the web page.



The procedure below describes how to add a custom header to the cobranding package. Follow the steps, and then deploy the cobranding package to enable the custom headers.

Prerequisites

You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).

Procedure

1. Create a custom header to add to the Spotfire environment. You can copy and edit the [WebHeader template file](#).
2. You can add a logo to the custom header, if desired. Use `/[%AppPath%]/`, e.g., `` and include an image in the cobranding package.



If you also want to change the product logo displayed on, for example, the login page, this is controlled in the `cobranding.config` file. Locate the section `<navigationLogo>libraryNavigationLogo.png</navigationLogo>`. Make sure that the file name inside the `<navigationLogo> ... </navigationLogo>` tags matches the name of the image file you want to use for the logo. The image file should have the dimensions 190x33 pixels.

3. Add the custom header HTM file to the local folder where the `cobranding.config` file is located.
4. Open the `cobranding.config` file in an XML editor or text editor.
5. Locate the section `<web> ... <header>WebHeader.htm</header> ... </web>`.
6. Make sure that the name in the `<header>` section matches the name of the HTM file that you want to use as a header.

Various header display options

If you want to display the custom header only for a certain UI, or if you want different web players to use different custom headers, the attribute `source` in the `Spotfire.Dxp.Worker.Web.config` file is used. The attribute can be set to any of the values `default`, `local`, and `none`.

By means of the different attributes, the following display options are possible (the first option is the default display option which you get when following the steps in the section above):

- The custom header is visible in the Server web administration pages as well as all web players. In this option, the value of the `source` attribute is `default` in the `Spotfire.Dxp.Worker.Web.config` file for each of the web players. (The default value of the `source` attribute is `default`.)
- To use the custom header is only in the Server web administration pages:
 1. Deploy the cobranding package containing the custom header in the default deployment area.
 2. For each web player, make sure `source` is set to `none` in its `Spotfire.Dxp.Worker.Web.config` file.
- To use the custom header only in all of the web client UIs:
 1. Check that no cobranding package is deployed in the default deployment area. (If a cobranding package is deployed in the default deployment area, the header will automatically be visible in the server UI.)
 2. Make sure that the cobranding package containing the custom header is deployed in a non-default deployment area.
 3. For each web player, make sure `source` is set to `local` in its `Spotfire.Dxp.Worker.Web.config` file.



Any web player that is deployed in the default deployment area will not get a custom header.

You can, if you want, make the custom header invisible in a separate web player. Then simply set `source` to `none` in its `Spotfire.Dxp.Worker.Web.config` file.



It is possible to use different custom headers in different web players by adding cobranding packages containing different headers to different deployment areas. Then set `source` to `local` in the `Spotfire.Dxp.Worker.Web.config` file for each of the web players.

The WebHeader template

You can copy and edit this `WebHeader.htm` template to create a custom header for the Spotfire Server web UI.

```
<div>
  <h1 style="background-color: #354856; padding: 10px 5px 10px 15px; margin: 0; font-
family: Arial, sans-serif; font-size: 15px; color: white; line-height: 1; font-weight: bold">
    Custom header
  </h1>
  <!-- 
  -->
```

```
</div>
```

Replacing the default Login page in the Spotfire Server web UI

You can create a custom login page using the cobranding functionality. This topic describes how to create a custom login for username/password authentication, or using third-party web authentication providers.

Prerequisites

- You have a `cobranding.config` file to work with. If you need one, see [Creating a cobranding.config file](#).
- In addition to the configuration file, you need the following files (links to templates and instructions are provided in the steps below):
 - An HTML file for the login page. The contents of this file are injected into the Spotfire Server web interface (for example, `custom-login-app-example.html`).
 - A JavaScript glue module. This file is the link between your HTML file and the Custom Login Page API (for example, `custom-login-app-example.js`).
 - A CSS file (Optional. For example, `custom-login-app-example.css`).

Procedure

1. Open the `cobranding.config` file in an XML editor or text editor.
2. Locate the section `<web>...</web>` and add the row `<customLoginPage>custom-login-app-example.html</customLoginPage>`, pointing to your custom html page.

Example:

```
<cobranding>
  ....
  <web>
    <customLoginPage>custom-login-app-example.html</customLoginPage>
  </web>
  ....
</cobranding>
```

3. Write or update your custom login HTML page. See [The custom-login-app-example.html template](#) on page 28 for an example to begin with.



The HTML file must not contain the `<html>`, `<head>` or `<body>` tags. The contents of this file are injected into the Spotfire Server web interface, which is an HTML file that contains these elements.

In the HTML file, make sure you import the following external files:

- The JavaScript module glue script.
- Optionally, the external CSS file.



Because you upload these files using the cobranding mechanism, the path of these files is dynamic. To reference them correctly in your HTML, prefix them with `/[%AppPath%]/`.

Example:

```
<script type="module" src="/[%AppPath%]/custom-login-app-example.js"></script>
<link rel="stylesheet" href="/[%AppPath%]/custom-login-app-example.css" />
```

Continue by adding the HTML elements needed on your custom login page.

4. For handling the interaction between your custom login page and the Custom Login Page API, write a glue JavaScript module. See [The custom-login-app-example.js template](#) on page 29. In this file, you can listen to events in your login page, change the DOM, and call the API as needed.



The implementation must be based on JavaScript. Import the [Custom Login Page API](#) JavaScript module in your JavaScript glue module and instantiate it.

Example:

```
import TSSCustomLoginApi from "/spotfire/js-api/custom-login-api.js";
const customLoginApi = new TSSCustomLoginApi();
```

5. Optionally, create or update the external CSS file. See [The custom-login-app-example.css template](#) on page 31.
6. Place your files (the configuration file, the HTML page, the glue script, and, optionally, the CSS file) together in a directory on your computer, and then follow the instructions in [Creating and deploying a cobranding package](#) on page 33.

The custom-login-app-example.html template

You can copy and edit this `custom-login-app-example.html` template to create a custom login page for the Spotfire Server web UI.

See [Replacing the default Login page in the Spotfire Server web UI](#) on page 27 for more information.

```
<script type="module" src="[%AppPath%]/custom-login-app-example.js"></script>
<link rel="stylesheet" href="[%AppPath%]/custom-login-app-example.css" />
<div class="container">
  <h2>ACME Login</h2>
  <div class="form">
    <input
      id="username"
      type="text"
      placeholder="Username"
      tabindex="1"
      name="username"
      autofocus="autofocus"
      autocomplete="off"
      autocapitalize="off"
      autocorrect="off"
      spellcheck="false"
    />
    <input
      id="password"
      type="password"
      placeholder="Password"
      tabindex="2"
      name="password"
      autocomplete="off"
      autocapitalize="off"
      spellcheck="false"
    />
    <button onclick="doLogin()">Login</button>
    <div class="form-group">
      <input id="eula" name="eula" type="checkbox" />
      <label for="eula">
        I accept the
        <a href="http://spotfire.com/">terms of service</a>
      </label>
    </div>
    <div id="remember-group" class="form-group hidden">
      <input id="remember" name="remember" type="checkbox" />
      <label for="remember">Remember me</label>
    </div>
    <div id="error" class="form-group hidden"></div>
  </div>
  <div id="wap-container" class="hidden">
    <hr />
    <ul id="wap-list"></ul>
  </div>
```

</div>

The custom-login-app-example.js template

You can copy and edit this `custom-login-app-example.js` template to create a custom login page for the Spotfire Server web UI.

See [Replacing the default Login page in the Spotfire Server web UI](#) on page 27 for more information.

```
import TSSCustomLoginApi from "/spotfire/js-api/custom-login-api.js";
const customLoginApi = new TSSCustomLoginApi();

customLoginApi.getServerManifest().then((serverInfo) => {
  // Do not show login page if user is already logged in.
  if (!serverInfo.showLoginPage) {
    // You can redirect to the URL set in the 'targetUrl' query parameter
  }

  // Remember
  const rememberGroup = document.getElementById("remember-group");
  if (rememberGroup && serverInfo && serverInfo.allowSaveInformation) {
    rememberGroup.classList.remove("hidden");
  }

  // Web Authentication Providers
  addWebAuthProviders(serverInfo.webAuthProviders);
});

function doLogin() {
  const username = document.getElementById("username").value;
  const password = document.getElementById("password").value;
  const remember = document.getElementById("remember").checked;
  const eula = document.getElementById("eula").checked;

  // Optional payload that can be passed to the server. Key prefix must start
  // with sf_custom_login_, otherwise they will not be sent in the request.
  const optionalPayload = {
    sf_custom_login_extra_string: "My own custom login value",
    sf_custom_login_extra_boolean: true,
    invalid_key: "keys need to start with sf_custom_login_",
  };

  if (!username || !password) {
    showErrorMessage("Please specify both username and password!");
    return;
  }

  if (!eula) {
    showErrorMessage("You must accept the terms of service!");
    return;
  }

  removeErrorMessage();

  customLoginApi.login(
    username,
    password,
    remember,
    null,
    loginError,
    optionalPayload
  );
}

function loginError(response) {
  if (response.status === 401) {
    showErrorMessage("Wrong username or password.");
  } else if (response.status === 0) {
    // Getting a status of 0 when server could not be contacted.
    showErrorMessage("Server could not be contacted");
  } else {
    // If anything else then unauthorizd, display error message.
    showErrorMessage("Internal Server Error");
  }
}
```

```

function showErrorMessage(message) {
  if (!message) {
    return;
  }

  const errorElement = document.getElementById("error");
  if (!errorElement) {
    return;
  }

  errorElement.innerText = message;
  errorElement.classList.remove("hidden");
}

function removeErrorMessage() {
  const errorElement = document.getElementById("error");
  if (!errorElement) {
    return;
  }
  errorElement.classList.add("hidden");
}

function addWebAuthProviders(webAuthProviders) {
  if (!webAuthProviders || webAuthProviders.length === 0) {
    return;
  }

  const ul = document.getElementById("wap-list");
  webAuthProviders.forEach((wap) => {
    const li = document.createElement("li");
    const button = document.createElement("button");
    button.textContent = "Login with " + wap.label;
    button.setAttribute("data-provider", wap.providerName);
    button.onclick = function (event) {
      const provider = event.target.dataset.provider;
      if (provider) {
        customLoginApi.redirectToWebAuthProvider(provider);
      }
    };

    li.appendChild(button);
    ul.appendChild(li);
  });

  // Show
  document.getElementById("wap-container").classList.remove("hidden");
}

document.getElementById("login-button").addEventListener("click", doLogin);

window.doLogin = doLogin;

```

The Custom Login Page API

The Custom Login Page API is a JavaScript module called `custom-login-api.js`, which exports the `CustomLoginApi` class. Instantiate this class to use the available functions.

login

```

/**
 * Logs a user in the Spotfire Server by taking the username, password and any optional
 * parameters supplied
 * and posts them to the Spotfire Server along with remember me value if that
 * setting is specified.
 *
 * @param {string} username The username
 * @param {string} password The password
 * @param {boolean} saveLoginInformation The actual value of
 * remember me
 * @param {function} [successCallback] Optional callback function called when the user has
 * been logged in successfully.
 * @param {function} [errorCallback] Optional callback function called when the login request
 * failed. If not provided, in case the request fails, it will throw an error.
 * @param {object} [optionalPayload] Optional payload that is encoded and passed

```

```

* along in the request body. All keys must be prefixed with sf_custom_login_.
* These parameters will be available to PostAuthenticationFilter
* implementations as a map attribute through the AuthenticationContext
* (details can be found in the Javadoc for AuthenticationContext).
*
* @since 11.4.0
*/

```

redirectToWebAuthProvider

```

/**
 * Redirects to the web authentication (e.g. OpenID Connect) provider. Will check targetUrl
 * query parameter and pass that along to the web auth provider.
 *
 * @param {string} providerName The name of the provider.
 * @param {function} [errorCallback] Optional error callback that executes if the call to
 * getting the authentication endpoint would fail.
 *
 * @since 11.4.0
 */

```

getServerManifest

```

/**
 * Fetches the Spotfire Server manifest information
 * @param {function} [errorCallback] Optional error callback that executes if the call getting
 * the Spotfire Server manifest information fails. If not provided, in this case, it will throw
 * an error.
 *
 * @since 11.4.0
 */

```

The custom-login-app-example.css template

You can copy and edit this `custom-login-app-example.css` template to create a custom login page for the Spotfire Server web UI.

See [Replacing the default Login page in the Spotfire Server web UI](#) on page 27 for more information.

```

body {
  margin: 0;
  padding: 0;
  width: 100%;
  height: 100%;
  display: flex;
  justify-content: center;
}

.container {
  width: 100%;
  max-width: 20rem;
  margin-top: 9rem;
}

.form input {
  margin: 0;
}

.form input[type="text"],
.form input[type="password"] {
  width: 100%;
  margin-bottom: 1rem;
  padding: 0.5rem;
}

.form input[type="checkbox"] {
  margin-right: 0.5rem;
}

```

```
.form button {  
  padding: 0.25rem 2rem;  
  margin-bottom: 2rem;  
}  
  
.form-group {  
  width: 100%;  
  display: flex;  
  margin-bottom: 1rem;  
  align-items: center;  
}  
  
.hidden {  
  display: none;  
}
```


Creating and deploying a cobranding package

To use the cobranded version of Spotfire, you must create and deploy a package containing the updated `cobranding.config` XML file together with any images and custom files, such as the custom header HTML file, or custom login files, that are needed.

Prerequisites

- You must have the following in a local folder:
 - An edited `cobranding.config` file.
 - All the images you plan to include.
 - All the files you need. For example, if you plan to use a custom header, a custom header .html file.
- For cobranding to work on a Linux system, cabextract must be installed on the computers running the Spotfire Server and the node managers (if you have node managers running on Linux).
- To use the cobranding package with a Web Player running on Linux as well as with Spotfire Analyst and Web Player on Windows, the package must use `IntendedPlatform="UNIVERSAL"` and `TargetFramework="ANY"`. Read more about building packages for Linux in the [Spotfire Developer Documentation](#).



You must use the Spotfire Package Builder to create the cobranding package. If you intend to build packages that work with the Web Player on Linux, you must use the [Package Builder console](#), `Spotfire.Dxp.PackageBuilder-Console.exe`, and not the graphical Package Builder tool.

Procedure

1. Download and unpack the Spotfire Developer software, which is available under Spotfire Server on [TIBCO eDelivery](#).
For download instructions, see the [Spotfire Developer documentation](#).
2. Start Spotfire Package Builder.
3. Click **Manage**, and in the Manage Configurations dialog box, click **Add**.
4. Enter a name for the configuration, and then click **OK** twice.
5. In the main Package Builder dialog, click **File > Add Spotfire Distribution**, to add a Spotfire executable for debugging the cobranding package. Browse to the `Modules` folder for your Spotfire Analyst installation. For example:
 - For a shared computer setup, this folder is usually located at `C:\Program Files (x86)\Spotfire\<version number>\Modules`.
 - For a single user setup, this folder is usually similar to `C:\Users\<username>\AppData\Local\Spotfire\<version number>\Modules`.
 (Note that the `Modules` folder is a hidden folder in some Windows configurations.)
6. Click **Select Folder**.
7. In the main Package Builder dialog, expand **Spotfire Distribution** and select the check box of the imported client version.
8. Click **Add**, leave the selected option **From folder**, clear the **Source code folder** check box, and then browse to and select the folder that contains all your cobranding project files.
9. Click **Select Folder** and then click **Next**.

10. On the next page, select **Create a module.xml file** and click **Next**.
If the file already exists, for the `File exists` message, click **Yes**.
11. On the next page, select both the **cobranding.config** check box and the words "cobranding.config" and then, under **Public resource name**, enter the text `cobranding.config`.
12. Select all the files to include in the package, including the `cobranding.config` file, and then click **Next**.
13. In the next dialog, give the project a name.
14. Under **Intended client**, select one of the following and then click **Finish**:
 - **Spotfire Installed Client** to cobrand just Spotfire Analyst.
 - **Spotfire Web Client** to cobrand Spotfire Web Player.
 - **Spotfire Any Client**
15. In the main Package Builder dialog, select the new project and then click **File > Build package file**.
16. In the Save Package dialog, select a location for the package file and click **Save**.
17. Click **Validate and Save**.
18. Deploy the cobranding package to Spotfire Server.
For instructions, see [Adding software packages to a deployment area](#) in the Spotfire® Server Installation and Administration guide.



The cobranding package must be deployed on the default deployment area.

What to do next

- If updates are not applied automatically to the default deployment area, make sure that end users accept the update when connecting to the server.
- If you are adding a custom header to the web client, you must update the Web Player services on each node of the collective. For instructions, see "Updating services" in the Spotfire Server help.

Spotfire Documentation and Support Services

For information about the Spotfire® products, you can read the documentation, contact Spotfire Support, and join the Spotfire Community.

How to Access Spotfire Documentation

Documentation for Spotfire and TIBCO products is available on the [TIBCO Product Documentation](#) website, mainly in HTML and PDF formats.

The website is updated frequently and is more current than any other documentation included with the product.

Spotfire Documentation

The documentation for all Spotfire products is available on the [Spotfire Documentation](#) page. This page takes you directly to the latest version of each document.

To see documents for a specific Spotfire product or version, click the link of the product under 'Other versions', and on the product page, choose your version from the top right selector.

Release Version Support

Some release versions of Spotfire products are designated as long-term support (LTS) versions. LTS versions are typically supported for up to 36 months from release. Defect corrections will typically be delivered in a new release version and as hotfixes or service packs to one or more LTS versions. See also <https://spotfi.re/lts>.

How to Contact Support for Spotfire Products

You can contact the Support team in the following ways:

- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the support portal at <https://spotfi.re/support>.
- For creating a Support case, you must have a valid maintenance or support contract with Cloud Software Group, Inc. You also need a user name and password to log in to <https://spotfi.re/support>. If you do not have a user name, you can request one by clicking **Register** on the website.

System Requirements for Spotfire Products

For information about the system requirements for Spotfire products, visit <https://spotfi.re/sr>.

How to join the Spotfire Community

The Spotfire Community is the official channel for Spotfire customers, partners, and employee subject matter experts to share and access their collective experience. The Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from Spotfire products. In addition, users can submit and vote on feature requests from within the [Ideas Portal](#). For a free registration, go to <https://spotfi.re/community>.

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. ("CLOUD SG") SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, "INCLUDED SOFTWARE"). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

Spotfire, the Spotfire logo, TERR, and TIBCO are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries. A list of Cloud SG's trademarks and trademark guidelines is available at <https://www.cloud.com/legal>.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG's Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the Apache License, Version 2.0, which is available at <https://www.apache.org/licenses/LICENSE-2.0> and reprinted in the Addendum below.

Copyright (c) Christian Robertson / Google, Roboto font.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the "readme" file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.tibco.com/patents>.

Copyright © 1994-2024 Cloud Software Group, Inc. All Rights Reserved.

Addendum to Legal and Third-Party Notices

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS